RoboCupRescue 2006 - Robot League Team
Team RFC Uppsala (Sweden)

Daniel Styrström, Mattias Holmqvist

Uppsala University
Department of Information Technology
Box 337, SE-751 05 Uppsala, Sweden
jakob.carlstrom@it.uu.se

http://www.robocup.it.uu.se/itp2006/

Abstract. This document describes the autonomous robot team RFC Uppsala and its contribution to RoboCup2006. Our system performs rescuing missions in a fully autonomous manner. Some of the more innovative features are the 3D map drawing, advanced AI and robust network communication with ad-hoc capabilities. One of the most interesting features described here is the laser scanner, which greatly improves the localization and mapping (SLAM). System management is conducted via a user-friendly GUI. We also give our thoughts and ideas on how the system performs in a realistic disaster situation.

Introduction

Team RFC Uppsala's contribution to RoboCup2006 consists of a multi-platformed team of two different kinds of robots. Both robot platforms are fully autonomous but have quite different roles in a rescue mission. Cooperation is greatly emphasized and the robots' physical characteristics naturally divide tasks between them. Gullfaxe, a wheel-based robot is based on the last year's Ringhorne design and has the role of quickly covering large areas with relatively flat terrain. As a complement, Sleipner, a track-based robot with better movement capabilities is designed to take care of rough terrain. The goal with this approach is to cover as many different disaster sites as possible without wasting valuable manpower.

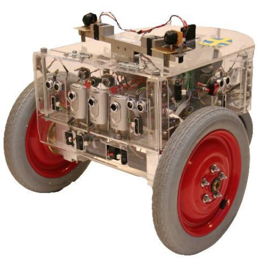

Fig. 1. The wheel-based Ringhorne robot from last year

The team of constructors consists of 11 senior MSc students and 4 advisors with different areas of expertise. We are working together with other senior staff from the university. Uppsala University has a long history in both RoboCup Middle Size Soccer League and Rescue Robot League.

1. Team Members and Their Contributions

In Table 1, we list the team members and their technical contributions in the project. We also list the senior advisors that have been greatly involved in the project.

Table 1. Team members and their technical contributions

Team member	Technical contribution
Hampus Berg	Project leader, simulation and Player programming.
Mattias Holmqvist	Technical advisor for Sleipner, networking and AVR programming.
Jakob Carlström	Senior advisor
Mikael Carlsson	Senior advisor
Carl-Johan Larsson	AI and reinforcement learning.
Ibrahim Lind	Simulation, GUI and Player programming.
Klas Pettersson	3D SLAM and AI.
Olof Rensfelt	Senior advisor
Johan Ringdahl	Networking and movement primitives.
Magnus Rundlöf	Lead programmer and 3D SLAM.
Daniel Styrström	Technical advisor for Gullfaxe, electronics and AVR programming.
Edvard Sedvall	Electronics and hardware.
Mattias Wiggberg	Senior advisor
Daniel Wikberg	Electronics and hardware.
Monika Zajaczkowska	Simulation, GUI and Player programming.

2. Operator Station Set-up and Break-Down (10 minutes)

The entire robot system is set up easily due to the robots' simple one-piece constructions. Sleipner is based on the lightweight Tarantula toy robot and is therefore easily transported. The wheel-based robot is slightly heavier but set-up is conducted in the same way as for the track-based robot.

Since the robots are fully autonomous, the operator station doesn't need to be manned during operation. However, it is possible to control the robot system manually if desired.

To manage the robot, the operator only needs access to a computer with a wireless 802.11a connection and the robot interaction software installed.

3. Communications

The robot system uses the IEEE 802.11a standard for wireless communication between the robots and the operator station. Due to expected radio interference, TCP Westwood is used for control messages. This is an enhanced TCP standard, especially good for lossy links [2]. UDP is used for streaming audio and video.

In the case of communication failure, the robots store as much sensor information as possible. If communication has not been re-established and the robot runs out of memory, a backtracking mechanism attempts to find radio coverage.

For the internal communication between the sensors' microcontrollers and the main processor, the CANbus is utilized [1].

Table 2. Desired communications channels.

Frequency	Channel/Band	Power (mW)
5.0 GHz - 802.11a	Primary: 36/Low 40/Low 44/Low 48/Low	40mW

4. Control Method and Human-Robot Interface

Since the robot system works in a fully autonomous manner, no interaction is needed for a rescue mission except set-up of the system. However, it is possible to take control over the robots via a GUI when desired. It's also easy to manually manipulate the map to correct obvious errors.

The main purpose of the control interface is to function as a surveillance system. All important information is sent to the control interface from the robots whenever a communications channel is available. This information is presented in the GUI on the operator station in the form of video feeds, sensor displays etc.

If manual control over the system is desired, only one operator is needed for controlling all the robots.

5. Map generation/printing

The robots work together by distributing the work load of map construction. Also, map data is sent between the robots to construct a full view of the discovered terrain. This is in turn be collected by the operator station for visualization and modification in the GUI.

Sleipner robots use 3D SLAM (Simultaneous Localization And Mapping) algorithms for map generation since the robot works in rough terrain with significant height variations [4].

Since Gullfaxe's main task in a rescue mission is to quickly cover large flat areas it has the possibility to distribute the 3D-SLAM calculations to an external server. If map data is available from other robots this will be used in navigational purposes.

Printing of the final map is easily done with a portable printer connected to the operator station.

6. Sensors for Navigation and Localization

All sensors are connected to the robot's central computer through a high speed CAN bus. The Player server is used to get a clear and structured interface to the hardware [5].

6.1 Laser Scanner

For scanning the terrain, each robot uses a movable laser scanner. In its original format, the scanner has a horizontal coverage of 240°. The laser also rotates on a horizontal axis, making its signal represent a view of the world in 3D. This makes it possible to discover different obstacles and also to generate a 3D map, necessary for secure movement in more rough terrain. The laser scanner also makes it possible to determine the position of the robot, based on the generated 3D map. For these functionalities, 3D SLAM algorithms are implemented [4].

Fig. 2. Hokuyo URG-04 LX Laser scanner used for scanning the terrain.

6.2 Infrared Sensors

Infrared sensors are placed on all sides of the robot to identify and avoid any obstacles. Due to the short operating distance of the infrared sensors their main task is to investigate the robot's immediate surroundings. Data from these sensors is then used for altering the robot's course and to create an accurate map.

6.3 Ultrasonic Sensors

Ultrasonic sensors are used as a complement to the infrared sensors. The primary use for these sensors is to detect transparent solid objects (glass, plexiglass etc.) which can't be detected by the infrared sensors. Furthermore, these sensors have slightly longer operating range and at short distances they assist the infrared sensors in positioning.

6.4 Bumpers

All robots are equipped with bumper sensors in order to avoid collisions if the other sensors fail to detect an obstacle. They are also used to determine if a detected obstacle is movable or not by gently pushing potentially movable objects. This is made possible by the use of sensors that variably measures the pressure on the sensor.

7. Sensors for Victim identification

Both robots are using the same set of sensors for the purpose of victim identification. These sensors are connected to the rest of the system in the same manner as the localization sensors.

7.1 Pyro-electric sensors

Two pyro-electric sensors are used on each robot to detect emission of body heat from victims. The sensors are mounted on separate arms that move independently to improve the accuracy of localization of victims. This is an improvement from last years one-arm design.

7.2 Camera

A webcam is mounted in the front of each robot to detect movement and to take snapshots of found victims.

7.3 CO_2 sensor

A carbon dioxide sensor is mounted on all robots in order to tell if found victims are breathing or not.

8. Robot Locomotion

Our two robot designs have different movement behaviors and capabilities due to their completely different physical designs. However, both robot designs support rotation around their own axis.

8.1 Sleipner

Sleipner, our track-based robot has 4 tracks that can be moved forwards or backwards and the left and right sides move independently of each other. Furthermore, the tracks can be rotated on their axis to enable climbing over rough terrain and stairs. To aid the autonomous movement, the robot also has sensors for determining the angle of each individual track.

Fig. 3. The track-based robot in its unmodified state.

8.2 Gullfaxe

Gullfaxe, our wheel-based robot works with two driving wheels that move independently of each other and also has a passive rear wheel.

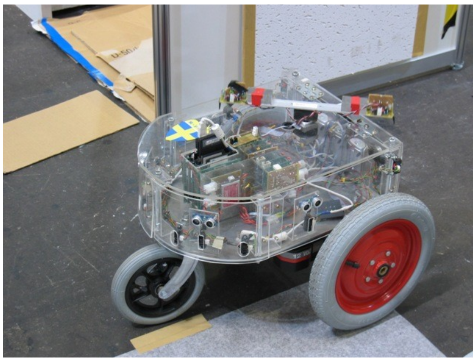
Fig. 4. The wheel-based robot from last years participation in RoboCup.

9. Other Mechanisms

Other mechanisms included in the robot system mainly consists of innovative software approaches described below.

9.1 Reinforcement learning

One mechanism that is used frequently in the robot system is reinforcement learning. This enables the robot to learn movement patterns and object discovery, important for both robot designs since they are both fully autonomous.

Due to more advanced movement capabilities, the track-based robot heavily depends on intelligent software and reinforcement learning has been a very useful tool.

9.2 Simulation Software

A simulation software supporting both robot designs has been implemented to aid in programming the AI and reinforcement learning mechanisms. The simulation engine is based on USARSim which is an open source simulation engine. USARSim is in turn based on the engine from the game Unreal Tournament [3].

The simulation engine simulates both robot designs in a 3D environment, making it possible to simulate the robots' movements in a realistic disaster site.

9.3 Modifications of the Tarantula

Because of the Tarantula's moderate robustness, some modifications are necessary. In order for additional components to fit Sleipner, the original chassis has been discarded. The added components add extra weight to the robot, which demands an overall stronger physical design.

9.4 Motor controllers

New motor controller cards have been developed by Lars Carlsson and Rickard Haglund at Uppsala University. Apart from the size reduction, a lot of other improvements have also been made. These include better debugging possibilities via a RS232 communication port instead of only LED's. Additional protection mechanisms such as temperature overload protection and voltage overload protection have also been added. Further on, a lot of different sensors can be connected to the controller card. In conclusion, it's a more flexible, safe, and expandable motor controller card.

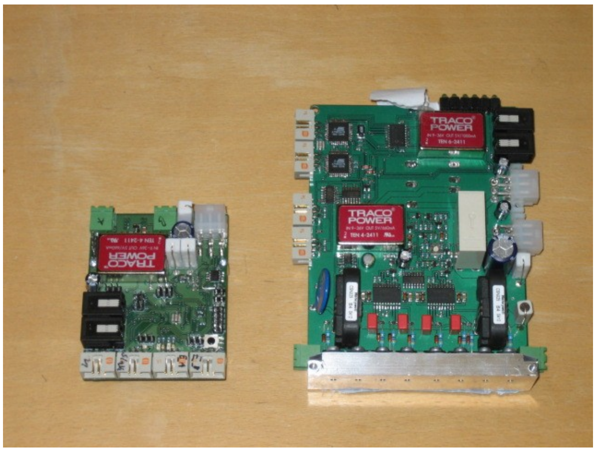

Fig. 5. The new motor controller card (left) and the old motor controller card (right).

9.5 Lamps

We use several GU4 lamps (Luxeon Star/O) to be able to take pictures in dark environments to aid the SLAM system and make remote operation possible.

10. Team Training for Operation (Human Factors)

Since the system is fully autonomous and a user-friendly GUI is provided, very little previous knowledge is required for operating the system. The only knowledge needed from the operators is fundamental knowledge of the displays in the GUI and the robots' individual capabilities.

For the purpose of training human operators, the simulator is a useful tool, since it is fully compatible with the GUI used by operators. Also, a test arena has been built in order to get an idea of the actual robots' individual behaviours.

11. Possibility for Practical Application to Real Disaster Site

Since the system is fully autonomous and also relatively cheap in production, it would be possible to distribute a large amount of robots over a vast disaster area to cover it more quickly. These type of robots can be used to conduct reconnaisance actions in disaster sites where it is dangerous to send in human personnel. Our system for scanning the terrain and even victim identification could be very useful in realistic disaster sites. However, it is still a challenging task to make autonomous robots move through difficult terrain without getting stuck or hurt themselves. This is one of the areas where improvement is needed for these kinds of systems to be applicable in realistic situations.

12. System Cost

The system costs for the different robot types given in the tables below are approximations. Since the robots are not yet fully developed, we cannot determine the costs precisely.

Table 3. System cost for Gullfaxe

No. Items	Module	Price (EUR)
1	WLAN Card - Orinoco Combocard Gold 802.11 a/b Cardbus	**70,52**
1	Webcam – Philips ToUcam Pro PCVC 740k	**65,32**
10	Devantech ultrasonic module	**358,00**
1	Ultrasonic control card	**150,00**
8	Sharp IR sensor	**38,94**
2	Front wheel	**31,35**
1	Rear wheel	**30,11**
2	DC-motor	**264,00**
2	Planetary gear	**234,30**
2	Pulse sensor	**90,20**
2	Assembly set	**7,92**
2	Motor controller cards	**236,00**
1	Frame work and material	**825,00**
2	Battery	**126,50**
1	Charger	**24,75**
2	Holds	**5,50**
8	LED	**40,00**
2	Luxeon Start lamps	**38,78**
2	OP-amplifiers	**1,28**
2	Rail-to-rail OP	**1,70**

2	Pyro-electric IR sensor (Nippon Ceramic)	**10,98**
2	Fresnel lens (Nippon Ceramic)	**10,03**
2	Stepper motor driver (Allegro)	**9,97**
1	Stepper motor	**56,32**
5	RFC CAN Cards	**32,18**
5	AVR Microcontrollers	**61,05**
5	CAN Controller	**6,60**
5	CAN Transceiver	**11,02**
5	Reset circuit	**8,18**
6	Optocoupler	**29,57**
1	USB PCMCIA Card Bus Adapter (2 port)	**18,07**
1	32MB SODIMM memory expansion	**19,72**
1	Hectatronic H6015 central computer	**221,54**
1	CompactFlash 128MB	**24,85**
1	Hectatronic H7006 CAN-card PC/104+	**38,50**
1	Cables	**33,00**
1	Assembly – testing	**16,50**
1	Hokuyo URG-LX04 Laser Scanner	**2200,00**
	Total robot cost	**5448,28**

Table 4. System cost for Sleipner

No. Items	Module	Price (EUR)
1	WLAN Card – Orinoco Combocard Gold 802.11 a/b Cardbus	**70,52**
1	Hokuyo URG-LX04 Laser Scanner	**2200,00**
3	AVR Microcontrollers	**37,00**
4	Sharp IR Sensor	**19,50**
4	Devantech ultrasonic module	**143,00**
1	Hectatronic CAN Card PC/104+	**38,50**
1	Hectatronic H6015 central computer	**221,54**
1	CompactFlash 128MB	**24,85**
1	32MB SODIMM expansion	**19,72**

2	Pyro-electric IR sensor (Nippon Ceramic)	**10,98**
1	Ultrasonic control card	**150,00**
1	USB PCMCIA Card Bus (2port)	**18,07**
4	CAN Controller	**5,28**
4	CAN Transceiver	**8,82**
4	RFC CAN Cards	**25,74**
4	Reset circuit	**6,54**
1	Tarantula toy robot (out of production).	**33,00**
2	Motor controller card	**236,00**
	Total robot cost	**3269,06**

References

1. Controller Area Network (CAN), Bosch, www.can.bosch.com/
2. TCP Westwood homepage, UCLA Computer Science Department - High Performance Internet Lab, http://www.cs.ucla.edu/NRL/hpi/tcpw/
3. USARSim homepage, http://sourceforge.net/projects/usarsim
4. Hartmut Surmann, Andreas Nüchter and Joachim Hertzberg, An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments, September 22, 2003
5. The Player/Stage project, http://playerstage.sourceforge.net